Capacity-aware back-pressure traffic signal control

Jean Gregoire Xiangjun Qian Emilio Frazzoli Arnaud de La Fortelle Tichakorn Wongpiromsarn

***Abstract*—The control of a network of signalized intersections is considered. Previous work demonstrates that the so-called back-pressure control provides stability guarantees, assuming infinite queues capacities. In this paper, we highlight the failing of current back-pressure control under finite capacities by identifying sources of non work-conservation and congestion propagation. We propose the use of a normalized pressure which guarantees work conservation and mitigates congestion propagation, while ensuring fairness at low traffic densities, and recovering original back-pressure as capacities grow to infinity. This capacity-aware back-pressure control allows to improve performance as congestion increases, as indicated by simulation results, and keeps the key benefits of back-pressure: ability to be distributed over intersections and $\mathcal{O}(1)$ complexity.**

I. Introduction

Congestion is one of the major problems in today's metropolitan transportation networks. Before investigating investments in order to enhance the capacity of the network, or policies to reduce the traffic load, one must wonder whether the network is used at its maximum capacity. Vehicle automation is expected to enable much more precise and intelligent coordination between vehicles, possibly reducing congestion [1]. However, automated cars are not currently ready for large commercial deployment. Human-driven cars can only be coordinated by traffic signals: more complex scheduling at intersections would require automation to be safe. That is why it is of high interest to study the theoretical maximum throughput of a network of intersections coordinated by traffic lights.

Traffic lights at intersections alternate the right-of-way of users (e.g., cars, public transport, pedestrians) to coordinate conflicting flows. A particular set of feasible simultaneous rights of way, called a phase, is decided for a certain period of time [2]. Controlling a traffic light consists of designing rules to decide which phase to apply over time.

Pre-timed policies activate phases according to a time-periodic pre-defined schedule. There is much previous work on designing optimal pre-timed policies. However, such policies are not efficient under changing arrival rates which require adaptive control. Most used adaptive traffic signal control systems include SCOOT [3], SCATS [4], PRODYN [5], RHODES [6], OPAC [7] or TUC [8]. These systems update some control variables of a configurable pre-timed policy on middle term, based on traffic measures, and apply it on short term. Control variables may include phases, splits, cycle times and offsets [2]. Such algorithms may differ in the way optimization is carried out (e.g., linear/dynamic programming, exhaustive enumeration) and in the modeling approach (e.g., queuing network model [9], cell transmission model [10], store-and-forward [11], petri nets [12]). Many major cities currently employ these systems which proved to be able to yield various benefits, including travel time and fuel consumption reduction, as well as safety improvements [13].

More recently, based on the seminal paper [14], feedback controls have been proposed both in the case of deterministic arrivals [15], or stochastic arrivals [16], [17]. Time is slotted and at every time slot, a feedback controller decides the phase to apply based on current queue length estimation. This requires real-time queues estimation, but it enables to be much more reactive than other traffic controllers and to have stability guarantees. Reference [14] introduced the so-called back-pressure control which computes the control to apply based on queue lengths, and can achieve provably maximum stability. This algorithm was originally applied to wireless communication networks [18], [19], and some effort has been required to apply the approach in the context of a network of intersections [16], [17]. A key feature of this algorithm is that it can be completely distributed over intersections, in the sense that it can be implemented by running an algorithm of complexity $\mathcal{O}(1)$, requiring only local information, at each intersection.

However, the strong assumption of current back-pressure traffic control algorithms is the unboundedness of queue capacities. Indeed, when the queue at the entry of an intersection grows so much that it reaches the upstream intersection, congestion will propagate: this is a non-negligible and easy to observe phenomenon. The phenomenon is commonly referred as blocking in queuing theory, and many blocking types can be considered [20]. In worst-case scenario, blocking results in deadlocks whose resolution can be of high complexity [21], [22]. An off-line optimization of a pre-timed policy is proposed in [9], [23]. The standard queuing network model with fixed service times of servers [24] is modified to account for blocking causing inter-queue interactions. The notion of effective service rate aims at accounting for both service and blocking. An expression of the blocking probability of each queue, i.e., the probability of the queue to be full, can be derived. The idea is to include in the optimized objective function penalties for high blocking probability. The method proved to be efficient to improve performance as congestion increases. However, it is for off-line optimization of fixed-cycle signals for a certain scenario (given arrival rates). In this paper, we aim at building a feedback control that can adapt on-line to varying situations. Some works applied to wireless communication networks have proposed feedback controls that can achieve maximum throughput under queue boundedness

constraints [25]. However, they suppose the absence of arrivals at internal nodes, cannot be easily implemented, and are thus not suitable for our application.

This paper proposes to keep the fundamental idea of back-pressure control, that is pressure computation at every node of the network, in order to keep the resulting key benefits: ability to be distributed over intersections and $\mathcal{O}(1)$ complexity. However, we propose to take into account the queue capacities for the computation of pressures. The idea is to normalize pressures, so that full queues all exert the same normalized maximal pressure independently from their capacity. Following the idea of [25], this normalization is expected to decrease the blocking probability.

The paper is organized as follows. Section II describes the phase-based queuing network model. Section III presents the current back-pressure traffic signal control of [17] and proves its lack of work conservation and its inability to avoid congestion propagation under finite queue capacities. Section IV proposes the use of normalized pressures and proves the benefits in terms of work-conservation and congestion mitigation. Simulations of Section V show the efficiency of the approach proposed in this paper and Section VI concludes and opens perspectives.

II. Model

A. Queuing network topology

The network of intersections is modelled as a directed graph of nodes $(N_a)_{a\in\mathcal{N}}$ and links $(L_j)_{j\in\mathcal{L}}$. Nodes represent roads with queuing vehicles, and links enable transfers from node to node. This is a standard queuing network model.

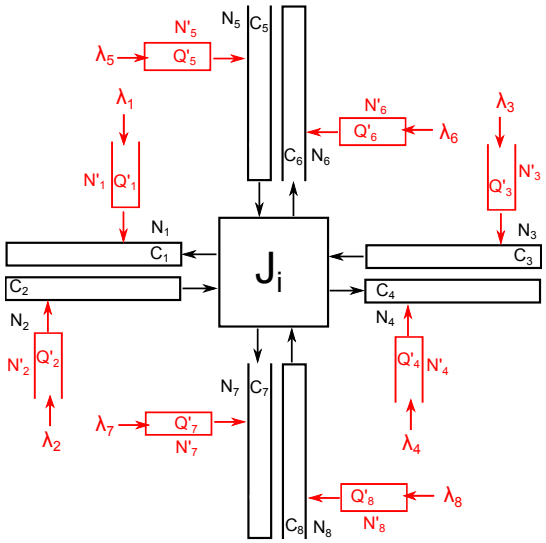


Fig. 1: A junction with 4 incoming nodes and 4 outgoing nodes which corresponds to the intersection depicted in Figure 2.

Every signalized intersection is modelled as a server managing a junction which consists of set of links. Junctions $(J_i)_{i\in\mathcal{J}}$ are supposed to form a partition of links. For every junction J, $\mathcal{I}(J)$ and $\mathcal{O}(J)$ denote respectively the inputs and the outputs of J. Inputs (resp.outputs) of junction J are nodes N such that there exists a link $L \in J$ pointing from (resp. to) N. The reader should consider the introduction of junctions in the model as an overlay of the queuing network model.

For the sake of simplicity, we do not represent links in the queuing network representation of Figure 1 and we assume that at every junction, there exists a link from any input to any output. If the link does not exist physically, the flow through it will be always zero.

Every server maintains an internal queue for every input/output, and server work enables to transfer vehicles from an input to an output of the junction. As standard in queuing network control, time is slotted, and each (time) slot $k \in \mathbb{N}$ maps to a certain period of time. Due to routing of vehicles, there are several queues at node N_a and $Q_{ab}(k)$ denotes the number of vehicles at node N_a in slot k waiting to leave node N_a for N_b. $Q_a(k) := \sum_b Q_{ab}(k)$ denotes the total number of vehicles waiting at node N_a.

B. Arrival and routing processes

Let $A_a(k)$ denote the number of vehicles that exogenously arrive at node N_a during slot k. We assume that the arrival process $A_a(k)$ is rate-convergent with rate λ_a which represents the expected number of arrivals per time slot at node N_a in the long-term. The arrival process is not controlled, it is an exogenous process. There are also endogenous vehicle transfers allowed by links at junctions. We let $f_{ca}(k)$ denote the number of vehicles leaving N_c for N_a during slot k. When a vehicle enters node N_a at slot k endogenously (originating from another node), or exogenously (inserted into the network through the arrival process), it increments one of the queues $Q_{ab}(k)$, unless it leaves the network at N_a. We assume that the ratio of vehicles added to $Q_{ab}(k)$ is rate-convergent with rate $r_{ab} \in [0,1]$. Rates r_{ab} represent the long-term routing ratios of vehicles entering N_a. Because some vehicles leave the network at N_a and are not added to some queue, the routing ratios do not necessarily sum to 1 and $1 - \sum_b r_{ab} \geq 0$ represents the exit rate at node N_a. The queue dynamics is as follows:

$$Q_{ab}(k+1) = Q_{ab}(k) - f_{ab}(k) + r_{ab}(k)\left(\sum_c f_{ca}(k) + A_a(k)\right), \tag{1}$$

where $r_{ab}(k)$ is rate-convergent with rate r_{ab}.

C. Phase-based control

At every time slot, the service offered by servers at junctions is controlled by activating a given signal phase p_i at every junction J_i from a predefined finite set of feasible phases $\mathcal{P}_i$. When phase p_i is activated during one slot, $\mu_{ab}(p_i)$ represents the maximum number of vehicles transferred from N_a to N_b during that slot. Figure 2 depicts the 4 typical phases of a 4 inputs/4 outputs junction. Each global phase $p = (p_i)_{i\in\mathcal{J}}$ results in a different service $\mu(p)$.

Two phenomena may affect the actual number of vehicles being transferred. First, only the vehicles which are currently at a queue at the beginning of the time slot can leave the queue during that slot. Second, and this is the phenomenon highlighted in this paper, above a certain queue length, a node is full and cannot accept vehicles any more.

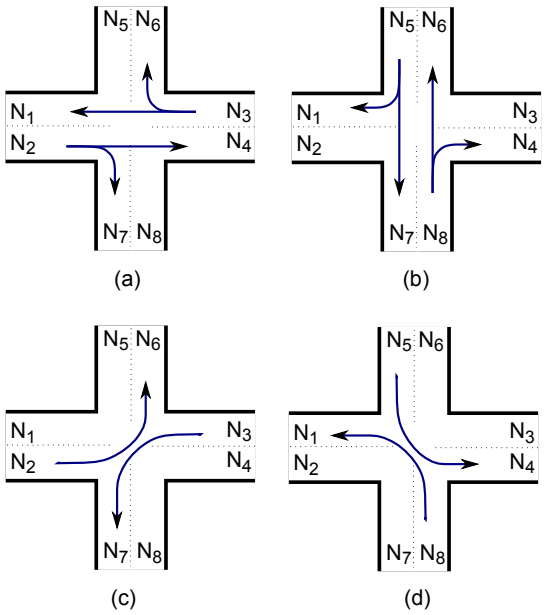


Fig. 2: A typical set of feasible phases at a junction. For phase (c), we have $\mu_{26}(p^{(c)}) > 0$ and $\mu_{37}(p^{(c)}) > 0$.

Let $\mathbf{p}(k)$ denote the phase control through time (the only controlled variable), the number of vehicles transferred from N_a to N_b during slot k is:

$$f_{ab}(k) \quad = \quad \delta(Q_b(k), C_b) \min(Q_{ab}(k), \mu_{ab}(\mathbf{p}(k)) \qquad (2)$$

The function $\delta(q, c)$ models blocking due to downstream congestion, and returns 1 if $q < c$, and 0 else. C_b is referred as the capacity of node N_b: it is the maximum queue length from which the node cannot accept vehicles any more. We say that node N_b is full at time slot k if $Q_b(k) \geq C_b$. The blocking phenomenon is illustrated in Figure 3. Note that this simple binary model is conservative because in reality, even if a node is full at the beginning of the time slot, some vehicles may be able to enter this node if the downstream junction gives the right-of-way to vehicles in that node. This effect will not be considered in this paper.

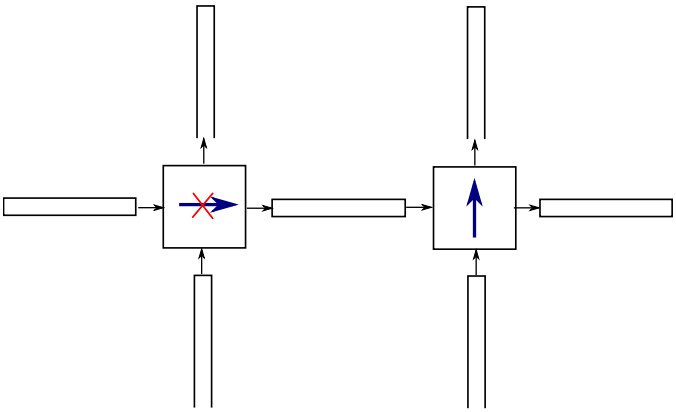


Fig. 3: Since the red-colored middle node is full, it cannot accept vehicles any more, even if the traffic light gives the right-of-way to vehicles going to this node.

III. Failing of back-pressure control under bounded queues constraints

A. Back-pressure control

Back-pressure control consists of a feedback control law that decides the phase to apply at every time slot based on current queues lengths. Let ϕ denote the control law, the phase control $\mathbf{p}$ satisfies at every time slot k:

$$\mathbf{p}(k) = \phi(Q(k)) \qquad (3)$$

where $Q(k)$ denotes the network state at slot k, that is the queues lengths matrix $[Q_{ab}(k)]_{a,b\in\mathcal{N}}$. The control law maps the current network state to the phase to apply. The back-pressure control law is defined component-wise for each junction as described by Algorithm 1. It only requires aggregated queues lengths $Q_a(k)$ and binary variables $d_{ab}(k) \in \{0, 1\}$ indicating the presence of vehicles waiting at N_a to leave N_a for N_b. The latter can be measured using loop detectors on the dedicated lanes at the entry of the junction.

Algorithm 1 Back-pressure control law at junction J_i

for $N_a \in \mathcal{I}(J_i) \cup \mathcal{O}(J_i)$ **do**
  $\Pi_a(k) \leftarrow P_a\left(Q_a(k)\right)$
end for
for $N_a \in \mathcal{I}(J_i), N_b \in \mathcal{O}(J_i)$ **do**
  $W_{ab}(k) \leftarrow d_{ab}(k) \max\left(\Pi_a(k) - \Pi_b(k), 0\right)$
end for
$\mathbf{p}_i(k) \leftarrow \arg\max_{p_i\in\mathcal{P}_i} \sum_{a,b} W_{ab}(k)\mu_{ab}(p_i)$
return Phase $\mathbf{p}_i(k)$ to apply in time slot k at junction J_i

The idea of back-pressure control is to compute pressures at every node of the network based on queue lengths and to allow flows with a high upstream pressure and a low downstream pressure, like opening a tap. Current back-pressure traffic signal control [17], [16] uses Algorithm 1 with linear pressure functions $P_a(Q_a) = Q_a$: the pressure exerted by a node equals its queue length. Algorithm 1 proceeds as follows:

- First, pressures at input/output nodes are computed. If linear pressures are used, the pressure exerted by node N_a at slot k is $P_a(Q_a(k)) = Q_a(k)$
- Then, the pressure difference associated to each flow from an input N_a to an output N_b of the junction is computed, it equals $\max\left(\Pi_a(k) - \Pi_b(k), 0\right)$, and it is multiplied by $d_{ab}(k)$ to account for the presence/absence of vehicles waiting at N_a willing to leave N_a for N_b.
- For each phase $p_i \in \mathcal{P}_i$, the total pressure release allowed by p_i is computed: it equals the sum of pressure differences through each link of the junction weighted by the flow of vehicles that can be transferred through the corresponding link when phase p_i is activated, that is $\sum_{a,b} W_{ab}(k)\mu_{ab}(p_i)$
- Finally, the returned phase $\mathbf{p}_i(k)$ is the phase p_i maximizing the weighted sum $\sum_{a,b} W_{ab}(k)\mu_{ab}(p_i)$.

This control law is proved to be stability-optimal under infinite capacities, i.e., the queuing network is stabilized for all arrival

rates that can be stably handled considering all control policies. The two key properties of back-pressure control are its ability to be distributed over junctions and its $\mathcal{O}(1)$ complexity.

Under bounded queues constraints, with linear pressure functions, pressure at N_a saturates at $P_a = C_a$ for $Q_a = C_a$. In the following, we show that this saturation at different levels for every node may result in a loss of work conservation and congestion propagation as presented in the sequel.

B. Loss of work conservation

First of all, the notion of work and work-conservation is defined in our context. We say that the server at junction J_i works during slot k if there are transfers through links of the junction during the slot. A control is work-conserving if the existence of an input N_a and an output N_b such that $Q_{ab}(k) > 0$ and $Q_b(k) < C_b$ is sufficient to ensure that the server of the junction works during slot k. A loss of work-conservation is clearly a sign of inefficiency. Due to limited queues capacities, back-pressure control under linear pressure functions is not work-conserving as stated in Theorem 1, with a concrete example depicted in Figure 4.

Theorem 1 (Loss of work-conservation under back-pressure)**.** *Under bounded queues constraints, back-pressure control is not work-conserving in the general case.*

Proof. Consider the network of Figure 4. Suppose that the middle junction has two feasible phases: p_{ab} with $\mu_{ab}(p_{ab}) > 0$, and p_{cd} with $\mu_{cd}(p_{cd}) > 0$. Suppose that at time slot k, $Q_b(k) = C_b$, $Q_a(k) > C_b$ and $Q_c(k) < Q_d(k)$. Then, $W_{ab}(k)\mu_{ab}(p_{ab}) > 0$ and $W_{cd}\mu_{cd}(p_{cd}) = 0$. Hence, the phase to apply at the middle junction computed by Algorithm 1 is p_{ab}. Since, $Q_b(k) = C_b$, Equation (2) implies that $f_{ab}(k) = 0$. As a result, the middle junction will not work, because the selected phase is p_{ab}, but due to downstream congestion, transfers to N_b are not feasible. However, choosing phase p_{cd} would have enabled the server to work. □

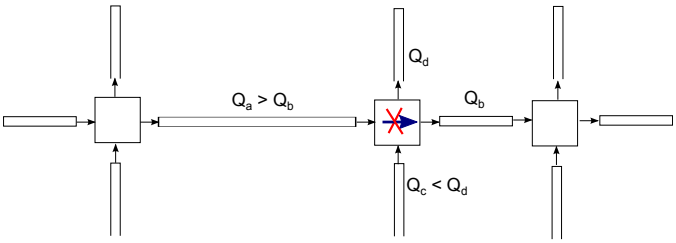


Fig. 4: Example of loss of work conservation of back-pressure control with linear pressure functions.

The proved loss of work conservation is an important property since the subsequent inefficiency results in congestion propagation as highlighted in the following.

C. Congestion propagation and deadlocks

As depicted in Figure 5, loss of work conservation may result in congestion propagation, both to the node which has the right-of-way but cannot empty because of downstream congestion, and to the node which has not the right-of-way.

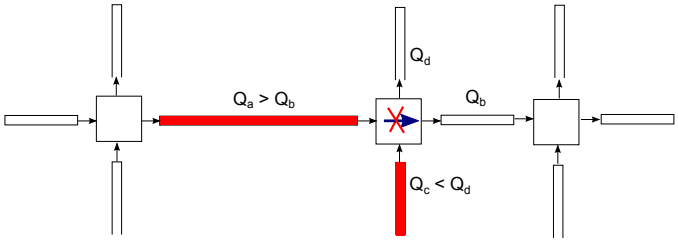


Fig. 5: Congestion propagation due to a loss of work conservation. Since N_b is full $f_{ab}(k) = 0$ and N_a is not emptied, so congestion propagates to N_a. Moreover, N_c is also not emptied because vehicles do not have the right-of-way under the selected phase, so congestion propagates to both nodes N_a and N_c.

In worst-case scenario, such congestion propagation can lead to deadlocks, as depicted in Figure 6.

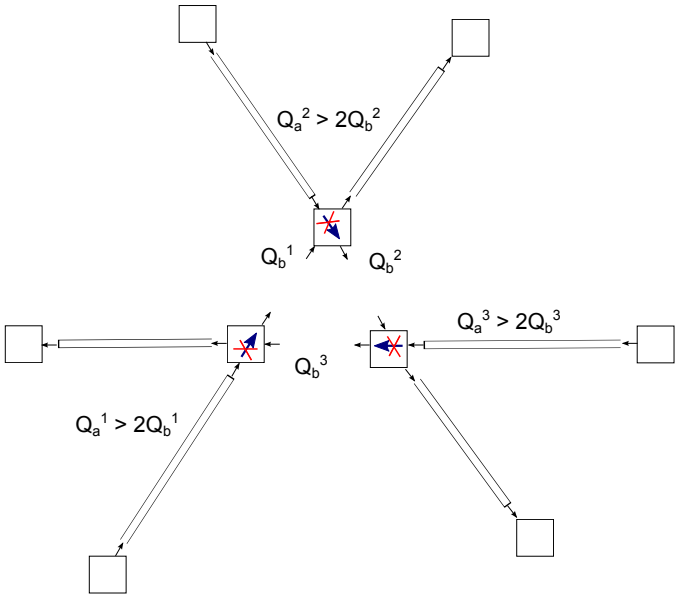


Fig. 6: A deadlock for back-pressure control. Q_a^1, Q_a^2 and Q_a^3 will be ever growing.

IV. Capacity-aware traffic control

The following presents our approach to take into account the limited queues capacities by using normalized pressures in back-pressure control to enforce work-conservation and mitigate congestion propagation. First of all, the reasons that motivate devising convex normalized pressures are introduced in the sequel.

A. Purpose of a convex normalized pressure and criteria

1) Purpose of a convex pressure: When a node approaches full occupancy, every additional vehicle is more and more problematic as the queue grows. That is why it makes sense to define a pressure such that the marginal pressure, i.e., the increase in pressure due to an additional vehicle, rises as the queue grows. This remark justifies the use of a convex pressure.

2) Purpose of a normalized pressure: When a node N_b is full, it does not make sense to have a node N_a such that $P_a - P_b > 0$, since the "tap" associated to the flow from N_a to N_b cannot be opened. That is why we propose to normalize the pressures so that any full node will exert a pressure that equals 1, while an empty node does not exert any pressure.

The simplest normalization that could be carried out would consist of using relative pressure functions $P_a(Q_a) = Q_a/C_a$. However, in order to have strictly convex pressures and to respect the fairness requirement proposed below, the normalization will be slightly more complex.

3) Requirement for fairness at low traffic density: At low traffic density, there is no reason to be unfair. Indeed, even a random choice of phases would stabilize the network. So, unfairness would not be justified by any global stabilization goal. Suppose that we use relative pressures functions $P_a(Q_a) = Q_a/C_a$, then an additional vehicle causes an increase in pressure of $1/C_a$, i.e., as high as capacity decreases. However, at low traffic density the marginal pressure should be uniform over nodes. We say that pressure functions $\{P_a(Q_a) : a \in \mathcal{N}\}$ are fair at low traffic density if:

$$\exists K > 0 : \forall a \in \mathcal{N}, \frac{dP_a}{dQ_a}(0) = K \tag{4}$$

4) Requirements for stability guarantees conservation: Finally, it is important to ensure that as capacities grow to infinity, the original back-pressure control is recovered, to take advantage of the stability guarantees in the context of infinite capacities. That is why a requirement for the pressure function $P_a(Q_a)$ is to be linear for $Q_a/C_a \to 0$. If pressure functions are fair, $P_a(Q_a) = KQ_a + o_a(Q_a/C_a)$ in Landau notation, and the pressure function is linear at low occupancy. As a result, if the queues are always much under maximum occupancy, i.e., if the infinite queues capacities assumption is valid, the back-pressure control and its stability guarantees are recovered.

Now we have presented criteria that should respect the modified pressure in order to be capacity-aware and fair at low traffic density, we propose in the following a convex normalized pressure which respects the above criteria.

B. Example of normalized pressure

The proposed pressure function should just be considered as an example of a pressure function fulfilling the presented requirements:

$$P_a(Q_a) = \min\left(1, \frac{\frac{Q_a}{C_\infty} + \left(2 - \frac{C_a}{C_\infty}\right)\left(\frac{Q_a}{C_a}\right)^m}{1 + \left(\frac{Q_a}{C_a}\right)^{m-1}}\right) \tag{5}$$

At low occupancy, the pressure at node N_a is linear: $P_a(Q_a) \simeq Q_a/C_\infty$, so pressure functions are fair and respect the requirement for stability guarantees conservation. The function is convex: the slope of the pressure increases as occupancy grows. Pressure over congestion threshold is normalized: $\forall a \in \mathcal{N}, \forall Q_a \geq C_a, P_a(Q_a) = 1$. The shape of pressure functions for two different capacities is depicted in Figure 7. One can observe that the pressure function leaves the initial linear behavior at lower occupancy as capacity decreases.

The parameters m and C_∞ determine the shape of pressure functions; m locates the transition from the linear regime, while C_∞ determines the slope of the pressure at low occupancy, and is such that a node which capacity is C_∞ will have a linear pressure. We assume that all capacities are lower than C_∞ and $m > 1$.

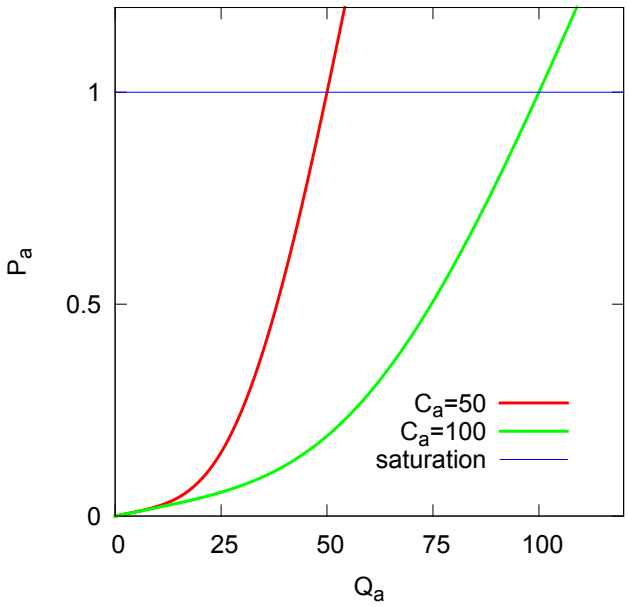


Fig. 7: Plot of the convex normalized pressure function with parameters $C_\infty = 500$ and $m = 4$ for two different congestion thresholds $C_a = 50$ and $C_a = 100$.

C. Work-conservation

As expected, pressure normalization enables to ensure work-conservation. This can be easily visualized in Figure 8 where one can observe that the deadlock of Figure 6 is resolved. In the following, we prove Theorem 2 which states work-conservation under convex normalized pressure.

Theorem 2 (Work-conservation under normalized pressures)**.** *Assume that pressure functions P_a are increasing functions taking values in $[0, 1]$, $P_a(0) = 0$ and $P_a(C_a) = 1$. Assume also that in case of equality, the* $\arg\max$ *of Line 7 in Algorithm 1 privileges phases p_i such that there exists a, b with $Q_{ab}(k) > 0$, $Q_b(k) < C_b$ and $\mu_{ab}(p_i) > 0$. Then, back-pressure control using normalized pressures is work-conserving.*

Proof. Suppose that back-pressure control using normalized pressure functions is not work-conserving. Then, there exists a server at a junction J_i which does not work during some slot k, while there exists a phase $\tilde{p}_i$ such that for some a, b, $\mu_{ab}(\tilde{p}_i) > 0$, $Q_{ab}(k) > 0$ and $Q_b(k) < C_b$.

Let $\mathbf{p}_i(k)$ denote the phase at junction J_i computed by Algorithm 1. If the server at the junction does not work during slot k under phase $\mathbf{p}_i(k)$, then for all c, d such that

$\mu_{cd}(\mathbf{p}_i(k)) > 0$ we have either $Q_{cd}(k) = 0$, or $Q_d(k) \geq C_d$ (otherwise, the server would work). In the first case, $d_{cd}(k) = 0$, and in the second case, $\Pi_d(k) = 1$. Since $W_{cd}(k) = d_{cd}(k) \max(\Pi_c(k) - \Pi_d(k), 0)$ and $\Pi_c(k) \leq 1$, we necessarily have $W_{cd}(k) = 0$. As a result, for phase $\mathbf{p}_i(k)$, we have $\sum_{cd} W_{cd}(k)\mu_{cd}(\mathbf{p}_i(k)) = 0$.

On the other hand, by positivity of flows and weights, we have $\sum_{cd} W_{cd}\mu_{cd}(\tilde{p}_i) \geq 0$. If $\sum_{cd} W_{cd}(k)\mu_{cd}(\tilde{p}_i) > 0$, it is absurd because $\tilde{p}_i$ should have been selected by Algorithm 1 instead of $\mathbf{p}_i(k)$. If $\sum_{cd} W_{cd}(k)\mu_{cd}(\tilde{p}_i) = 0$, it is also absurd because $\sum_{cd} W_{cd}(k)\mu_{cd}(\tilde{p}_i) = \sum_{cd} W_{cd}(k)\mu_{cd}(\mathbf{p}_i(k))$, and again, $\tilde{p}_i$ should have been selected by Algorithm 1 instead of $\mathbf{p}_i(k)$ because an equality holds but contrary to $\mathbf{p}_i(k)$, there exists for $\tilde{p}_i$ an input N_a and an output N_b with $\mu_{ab}(\tilde{p}_i) > 0$, $Q_{ab}(k) > 0$ and $Q_b(k) < C_b$ (see the second assumption in the theorem). □

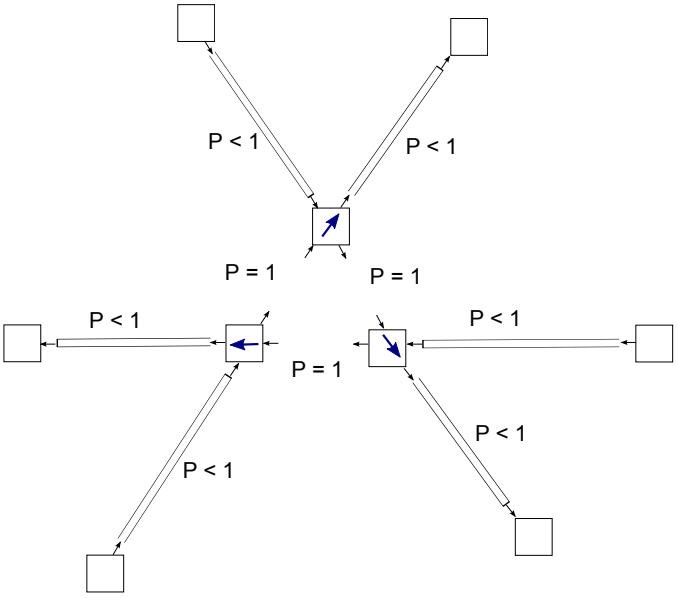


Fig. 8: Work-conservation and deadlock resolution using normalized pressure functions. The pressure at full nodes equals 1, while it is strictly lower than 1 at non-full nodes. As a result, the phase computed by Algorithm 1 enables to empty full nodes.

V. Simulation results

In this section, we compare the performance of our proposed capacity-aware back-pressure control with current back-pressure [17], [16] and a non-optimized fixed cycle traffic light given as reference.

A. Simulation setup

We have implemented our traffic signal control schemes on the top of the traffic simulator SUMO (Simulation of Urban MObility) [26]. SUMO is a widely recognized open-source traffic simulation package including a traffic simulator as well as supporting tools. The simulator is microscopic, inter- and multi-modal, space-continuous and time-discrete, providing a fair approximation of real world traffic scenarios.

We adopt a non-uniform network with several types of roads and intersections (Figure 9). All roads are bi-directional. Roads V2, V4, V6, V8, H1, H3, H5 and H7 possess only one lane on each direction while the rest of roads have two lanes. Close to the intersection, each road has an additional dedicated left-turn lane. Due to the difference in the number of lanes, there are four types of intersections (Figure 10). Each intersection has four phases, as described in Figure 2. The network is open as vehicles may leave and enter the system at all roads.

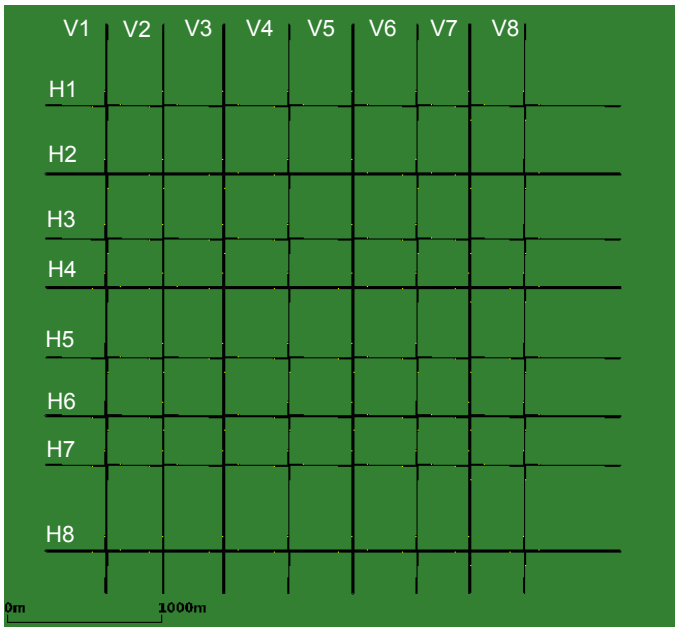


Fig. 9: The road network used for simulations.

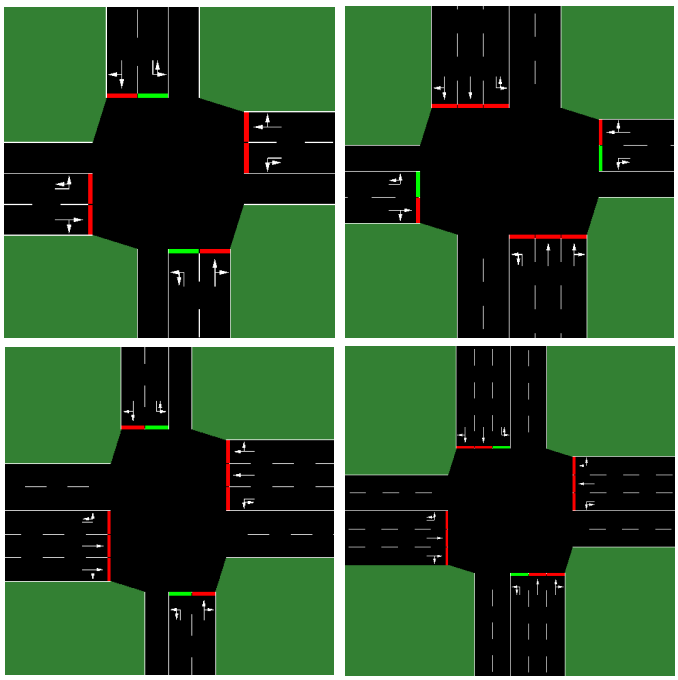


Fig. 10: Four intersection types depending on the number of lanes of incoming roads.

We generate traffic flows using *ActivityGen* available in

SUMO supporting tools. *ActivityGen* considers the road network as a city. It takes as inputs, in particular, the city population, the spatial distribution of the population, the working zones, and produces traffic flows with on/off peak patterns. In our network, we set the habitation area to northern city (area in blue rectangle in Figure 9) and the working zone resides in the southern city (area in red dashed rectangle in Figure 9). We design the test scenarios with a city population ranging from 10000 to 39000 people. We simulate the traffic during a typical 3-hour morning peak (7 am to 9 am). An exemplary histogram of vehicle arrivals in the morning peak is given in Figure 11. All Vehicles adopt the default vehicle model of SUMO. A python interface, called TraCI, is provided

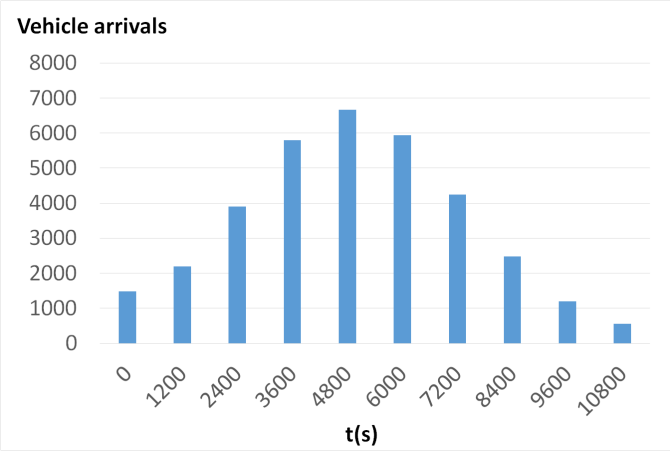


Fig. 11: Vehicle arrivals histogram during the morning peak hour (7 am to 9 am) when the city population is set to 33000.

in SUMO package which enables real-time traffic control. In particular, queues lengths can be retrieved by deploying loop detectors and the phase of traffic lights (more precisely the program) can be updated. We implemented control schemes as a python application. The application interacts with SUMO through TraCI to retrieve queue lengths and to control traffic lights. Three control schemes have been implemented:

1) Current back-pressure of [17], [16]: the duration of a time slot is set to 15 seconds, including 4 seconds of yellow phase.
2) Capacity-aware back-pressure: the capacity of roads is computed using the lane length retrieved through TraCI and the vehicle length and minimum gap. The length of the default vehicle in SUMO is 5 meters and the minimum gap is 2.5 meters. As a result, the capacity of a lane is its length in meters divided by 7.5. The time slot setting is the same as in back-pressure: 15 seconds, and the parameters of the convex normalized pressure of Equation (5) are $m = 2$ and $C_\infty = 200$.
3) Non-optimized fixed cycle control: traffic lights periodically switch the applied phase following the sequence (a)-(c)-(b)-(d) of Figure 2. A complete cycle lasts 60 seconds, in which phase (a) and (b) last 16 seconds and phase (c) and (d) last 6 seconds. The yellow time between two phases is set to 4 seconds. The duration of phases is fixed arbitrarily, it is not optimized.

A complete test run compares the three control schemes under 4 different populations. To ensure comparability, the same random seed is given to *ActivityGen* module for all simulations of a test run.

B. Results and analysis

Figure 12 depicts the evolution of the number of vehicles in the network under four population scenarios. We also measure periodically the total time spent by vehicles averaged over vehicles currently in the network, which is a good performance indicator, since users of the network want to minimize their travel time. The evolution of the total time spent is depicted in Figure 13. We observe that for a small population of 10000, all control schemes have a similar performance, even if the queue is slightly greater under the non-optimized fixed cycle control scheme. For a population of 27000, back-pressure and capacity-aware back-pressure have similar performance, while they both outperform the fixed-cycle control scheme. As population grows (leading to a growth of flows through the network), for a population of 33000 and 39000, capacity-aware back-pressure outperforms back-pressure control. Therefore, the performance gain of capacity-aware back-pressure mainly occurs at heavy load. Figure 14 presents a typical configuration that may occur under back-pressure control, while it is alleviated in capacity-aware back-pressure control. Figure 14(a) provides a bird eye view on the intersection between H5 and V4. H5 is a 2-lane main street and V4 has only one lane. V4 has reached its full capacity downstream. Figure 14(b) offers a closer look on the intersection. Since back-pressure control does not consider the capacity of roads, the left turn phase (phase (c) of Figure 2) of H5 is always activated because the queue difference of corresponding lanes is among the largest. However, it is inefficient and not work-conservative to activate this phase as no vehicle can join the downstream node of V4. Under the same situation, with capacity-aware back-pressure control scheme, the pressure on the downstream node of V4 will be 1, and phase (a) would be activated instead of phase (c) allowing vehicles on H5 going straight to continue their journey (work-conservation is recovered). This qualitative analysis on the example of Figure 14 is in accordance with quantitative results that show an increase in the average total time spent for back-pressure for a population of 33000 and a drift for a population of 39000 (see Figure 13).

VI. Conclusions and Perspectives

In this paper, we adapt current back-pressure control to take into account bounded queues constraints. The lack of work-conservation of current back-pressure control is proved, and identified as a source of congestion propagation through the network. This phenomenon is caused by pressure saturation at queues that have reached maximum capacity.

Normalized pressure functions are proved to ensure work-conservation and this property tends to indicate that congestion propagation will be mitigated. Simulations confirm the efficiency of the approach. It is remarkable that performance have been increased under bounded queues constraints as indicated by simulations, while the ability to distribute the

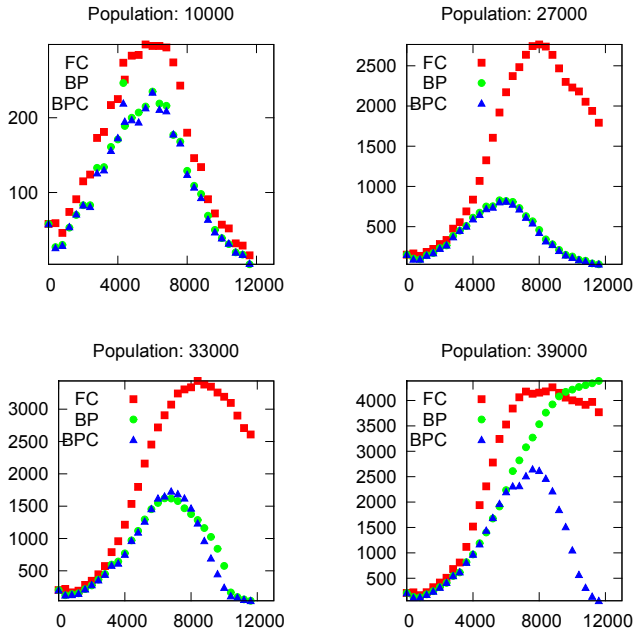


Fig. 12: Comparison of the total queue length in the network through time under different population scenarios: 10000, 27000, 33000 and 39000. Time is in seconds. FC refers to the fixed-cycle control scheme, BP to back-pressure and BPC to capacity-aware back-pressure.

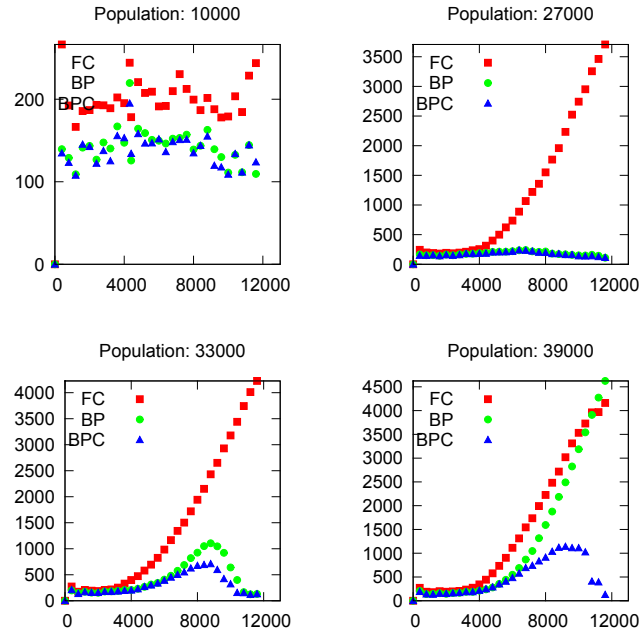
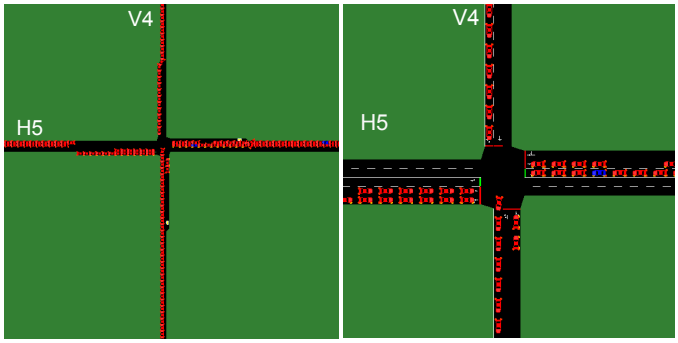


Fig. 13: Comparison of the average total time spent in the network through time under different population scenarios: 10000, 27000, 33000 and 39000. Time is in seconds. FC refers to the fixed-cycle control scheme, BP to back-pressure and BPC to capacity-aware back-pressure.

control over junctions and $\mathcal{O}(1)$ complexity properties have been conserved.

However, for very high arrival rates, and in particular above the capacity region, congestion will necessarily eventually propagate through the network. In this case, vast areas of the network will be congested and inter-junctions interactions due to blocking tend to indicate that phase control should be carried out on groups of junctions belonging to the same congested region. Nevertheless, this task is of high complexity due to the exponential complexity of inter-junctions interactions.

Finally, future works on back-pressure signal control should consider the feedback loop between traffic signal control and driver behaviour, and in particular driver routing choice. One can expect drivers at a junction to change their routing choice if the traffic light gives the right-of-way in favour of some particular output nodes due to traffic conditions. It is of high interest to take into account such behaviours, since they may stabilize or unstabilize the queuing network.



(a) (b)

Fig. 14: A typical inefficient configuration in back-pressure control when capacities are not considered. The screen shot is captured at time 7135 with a population of 39000

References

[1] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, March 2008.

[2] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2043–2067, 2003.

[3] P. Hunt, D. Robertson, R. Bretherton, and M. Royle, "The scoot on-line traffic signal optimisation technique," *Traffic Engineering & Control*, vol. 23, no. 4, 1982.

[4] P. Lowrie, "Scats, sydney co-ordinated adaptive traffic system: A traffic responsive method of controlling urban traffic," 1990.

[5] J.-J. Henry, J.-L. Farges, and J. Tuffal, "The prodyn real time traffic algorithm," in *Proceedings of the 4th IFAC/IFORS Conference on Control in Transportation Systems,*, 1984.

[6] P. Mirchandani and L. Head, "A real-time traffic signal control system: architecture, algorithms, and analysis," *Transportation Research Part C: Emerging Technologies*, vol. 9, no. 6, pp. 415–432, 2001.

[7] N. H. Gartner, "Opac: A demand-responsive strategy for traffic signal control," *Transportation Research Record*, no. 906, 1983.

[8] C. Diakaki, M. Papageorgiou, and K. Aboudolas, "A multivariable regulator approach to traffic-responsive network-wide signal control," *Control Engineering Practice*, vol. 10, no. 2, pp. 183–195, 2002.

[9] C. Osorio and M. Bierlaire, "A surrogate model for traffic optimization of congested networks: an analytic queueing network approach," *Report TRANSP-OR*, vol. 90825, pp. 1–23, 2009.

[10] H. K. Lo, E. Chang, and Y. C. Chan, "Dynamic network traffic control," *Transportation Research Part A: Policy and Practice*, vol. 35, no. 8, pp. 721–744, 2001.

[11] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos, "Store-and-forward based methods for the signal control problem in large-scale congested urban road networks," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 2, pp. 163–174, 2009.

[12] A. Di Febbraro, D. Giglio, and N. Sacco, "On applying petri nets to determine optimal offsets for coordinated traffic light timings," in *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*, pp. 773–778, 2002.

[13] S. Shepherd, "A review of traffic signal control.," 1992.

[14] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *Automatic Control, IEEE Transactions on*, vol. 37, no. 12, pp. 1936–1948, 1992.

[15] P. Varaiya, "The max-pressure controller for arbitrary networks of signalized intersections," in *Advances in Dynamic Network Modeling in Complex Transportation Systems*, pp. 27–66, Springer, 2013.

[16] P. Varaiya, "A universal feedback control policy for arbitrary networks of signalized intersections," tech. rep., 2009.

[17] T. Wongpiromsarn, T. Uthaicharoenpong, Y. Wang, E. Frazzoli, and D. Wang, "Distributed traffic signal control for maximum network throughput," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pp. 588–595, IEEE, 2012.

[18] M. J. Neely, *Dynamic power allocation and routing for satellite and wireless networks with time varying channels*. PhD thesis, LIDS, Massachusetts Institute of Technology, 2003.

[19] M. J. Neely, E. Modiano, and C. E. Rohrs, "Dynamic power allocation and routing for time-varying wireless networks," *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 1, pp. 89–103, 2005.

[20] H. G. Perros, *Queueing networks with blocking*. Oxford University Press, Inc., 1994.

[21] S. Kundu and I. F. Akyildiz, "Deadlock free buffer allocation in closed queueing networks," *Queueing Systems*, vol. 4, no. 1, pp. 47–56, 1989.

[22] J. Gregoire, S. Bonnabel, and A. De La Fortelle, "Priority-based coordination of robots." 29 pages, June 2013.

[23] C. Osorio and M. Bierlaire, "An analytic finite capacity queueing network model capturing the propagation of congestion and blocking," *European Journal of Operational Research*, vol. 196, no. 3, pp. 996–1007, 2009.

[24] P. P. Bocharov, C. D'Apice, and A. Pechinkin, *Queueing theory*, vol. Chapter 3. Walter de Gruyter, 2003.

[25] P. Giaccone, E. Leonardi, and D. Shah, "Throughput region of finite-buffered networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 18, no. 2, pp. 251–263, 2007.

[26] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, pp. 128–138, December 2012.